

REMARKS/ARGUMENTS

35 U.S.C. 112 based rejections

Claims 2, 21, 36, 49, 55, 61

Applicants have amended “the caller” to “a caller”, and submit that the amendments overcome the Examiner’s 35 U.S.C. 112 based rejections of claims 2, 21, 36, 49, 55, 61.

Claims 8, 27, 42

Applicants have amended “the Java Remote method Invocation (RMI) classes” to “Java Remote method Invocation (RMI) classes”, and submit that the amendments overcome the Examiner’s 35 U.S.C. 112 based rejections of claims 8, 27, 42.

35 U.S.C. 103(a) based rejections

The Examiner has rejected claims 1-65 under 35 U.S.C. 103(a) as being unpatentable over Jones (US 6134603) in view of Guthrie (US 6385661). Applicants have amended the independent claims and certain other claims, and traverse the rejections of claims 1-65.

Amended Independent claims 1, 20, 35

Amended independent claims 1, 20, 35 require implementing a remote method call, comprising:

- generating remote objects;
- generating at least one proxy object, where each proxy object corresponds to one remote object;
- including data from the remote object into the proxy object; and
- processing a call to a method on one proxy object by:
 - (i) determining whether the method is implemented locally;

(ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally; and

(iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

Applicants have amended the independent claims 1, 20, 35 to include the limitations that the call to a method on one proxy object is processed by:

(i) determining whether the method is implemented locally;

(ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally; and

(iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

The added limitations may be found in at least original claims 1, 20, 35, in FIG. 6: blocks 282, 286, 288, and 290 of the original Application, and the corresponding description of FIG. 6 in paragraph of the original Application. Applicants have removed the claim requirement of executing the method, wherein the method is one of a plurality of methods, wherein at least one of the plurality of methods comprises a local method including code to perform method operations on the proxy object without going to the remote object and at least one other of the plurality of methods comprises a remote method including code to perform method operations on the remote object.

The Examiner has rejected independent claims 1, 20, 35 under 35 U.S.C. 103(a) as being unpatentable over Jones in view of Guthrie. Applicants traverse, and submit that neither the cited Jones (col. 5: lines 29-32; col. 10: lines 38-45; reference numerals 304, 310, 312, 314, 316, 506, 508; col. 8: lines 8-16, 46-49, 55-67; col. 9: lines 52-67; col. 10: lines 5-14) nor the cited Guthrie (col. 6: lines 28-42), either alone or in combination, teach or suggest the claim requirements of processing a call to a method on one proxy object by:

- (i) determining whether the method is implemented locally;
- (ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally, wherein data the remote object is included into the proxy object; and
- (iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

The cited Jones discusses a proxy object (stub) on a client sending a hash value identifying a remote method to a server having a remote object via a RMI call. When the server receives the RMI call, the server identifies which method is being called using the received hash value. Nowhere does the cited Jones teach or suggest the claim requirements of processing a call to a method on one proxy object by:

- (i) determining whether the method is implemented locally;
- (ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally, wherein data the remote object is included into the proxy object; and
- (iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

The cited Guthrie discusses a software system that provides for dynamic generation of remote proxy classes at runtime. If an object is requested and a proxy does not exist for an object on a client system then the remote proxy for the requested object is generated. Nowhere does the cited Guthrie teach or suggest the claim requirements of processing a call to a method on one proxy object by:

- (i) determining whether the method is implemented locally;
- (ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally, wherein data the remote object is included into the proxy object; and

(iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

In addition, the Examiner has indicated that the cited Guthrie in col, 6, lines 28-42 discusses performing method operations on the proxy object without going to the remote object by indicating that proxies perform specific tasks such as controlling access to or communications with the object they represent and that proxies contain unique computer code to accomplish their assigned functions. Lines 28-42 of the cited Guthrie discusses the process of reflection, where according to the cited Guthrie, "the reflection process is only concerned with determining the public view of an object" and "proxies need to be reflections, or duplicates on the surface, of objects". Therefore, the cited Guthrie is discussing that the proxy object has similar interfaces to the remote object, but does not teach or disclose the claim requirements of processing a call to a method on one proxy object by:

- (i) determining whether the method is implemented locally;
- (ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally, wherein data the remote object is included into the proxy object; and
- (iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

Additionally, the Examiner has combined the teaching of the cited Jones and the cited Guthrie to reject the claims under 35 U.S.C. 103(a). According to the Manual of Patent Examining Procedure (MPEP) §2143.01 (8th Edition Revision 2, published May 2004, page 2100-131) "fact that references can be combined or modified is not sufficient to establish prima facie obviousness" and "the mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination." Applicants submit that the Examiner's proposed combination of the teachings of the cited Jones and the cited Guthrie is improper. According to the Examiner, the cited Guthrie's teachings would have provided the capability of dynamically generating remote proxy classes,

which is different from the claim requirements of processing a call to a method on one proxy object by:

- (i) determining whether the method is implemented locally;
- (ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally, wherein data the remote object is included into the proxy object; and
- (iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

Additionally, the motivations provided by the Examiner are for dynamically generating remote proxy classes which are different from the claims requirements of processing a call to a method on one proxy object by:

- (i) determining whether the method is implemented locally;
- (ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally, wherein data the remote object is included into the proxy object; and
- (iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

Therefore, there is no proper motivation either in the cited Jones or the cited Guthrie, nor has any proper motivation been provided by the Examiner to teach or suggest the claim requirements of processing a call to a method on one proxy object by:

- (i) determining whether the method is implemented locally;
- (ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally, wherein data the remote object is included into the proxy object; and
- (iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

For the above reasons, amended Independent claims 1, 20, 35 are patentable over the cited art.

Amended independent claims 48, 54, 60

The Examiner has rejected independent claims 48, 54, 60 under the same rationale as independent claim 1.

Claims 48, 54, 60 have been amended in a manner similar to the amendment to claims 1, 20, 35 and support for the claim amendments may be found in at least original claims 48, 54, 60, in FIG. 6: blocks 282, 286, 288, and 290 of the original Application, and the corresponding description of FIG. 6 in paragraph of the original Application.

Applicants submit that amended independent claims 48, 54, 60 are patentable over the combination of the cited Jones and the cited Guthrie because nowhere does the cited Jones or the cited Guthrie, either alone or in combination, teach or suggest the claims requirements of processing a call to a method on one proxy object by:

- (i) determining whether the method is implemented locally;
- (ii) executing a local method including code to perform method operations on the proxy object without going to the remote object, in response to determining that the method is implemented locally, wherein data the remote object is included into the proxy object; and
- (iii) executing a remote method including code to perform method operations on the remote object, in response to determining that the method is not implemented locally.

Additional reasons for the patentability of amended independent claims 48, 54, 60 are similar to the reasons provided by the Applicants for the patentability of claims 1, 20, 35.

For the above reasons amended independent claims 48, 54, and 60 are patentable over the cited art.

Dependent claims 2-19, 21-34, 36-47, 49-53, 55-59, 61-65

The Examiner has also rejected pending claims 2-19, 21-34, 36-47, 49-53, 55-59, 61-65. These claims are patentable over the cited art because they depend directly or indirectly on independent claims 1, 20, 35, 48, 54, or 60 which are patentable over the cited art for the reasons discussed above. Furthermore, the following of these claims provide additional grounds of patentability over the cited art for the reasons discussed below.

Amended Claims 2, 21, 36, 49, 55, 61

Claim 2, 21, 36, 49, 55, 61 depend on claims 1, 20, 35, 48, 54, 60 respectively, wherein executing one local method comprises processing data in the proxy object from the remote object, wherein remote object data is returned to a caller from the proxy object, wherein the remote object data is the data from the remote object included into the proxy object.

The claims have been amended to include the requirements that the remote object data is the data from the remote object included into the proxy object and the added requirements may be found in at least original claims 1, 20, 35, 48, 54, 60, and in at least pages 3-15 of the original Application..

The cited Jones (col. 9, lines 66 - col. 7, line 2) discusses invoking a method using the received parameter argument values in a RMI call, and returning the result of the method invocation to a client. Nowhere does the cited Jones teach or suggest the claim requirements that data from the remote object included into the proxy object is returned to the caller from the proxy object. While the cited Jones discusses returning a result to a client, nowhere does the cited Jones teach or suggest the claim requirement that data from the remote object included into the proxy object is returned to the caller from the proxy object.

For the above reasons amended claims 2, 21, 36, 49, 55, 61 are patentable over the cited art.

Amended Claims 11, 30, 45, 53, 59, 65

The Examiner has provided specific reasons for rejecting claim 11, and has indicated that claims 30, 45, 53, 59, are 65 have been rejected for reasons similar to the rejection of claim 11. Applicants traverse the rejection of claims 11, 30, 45, 53, 59, 65.

Claim 11 depends on claim 5 that depends on claim 1, and further comprises that the server communication object comprise a first server communication object, respectively, and wherein executing one remote method further comprises:

passing the remote method to the client communication object;

transmitting to the server communication object, with the client communication object, an invocation method specifying the remote method on one specified remote object to the server communication object;

determining, with the server communication object, whether the remote object specified in the received invocation method is accessible through a second server communication object; and

transmitting, with the first server communication object, the received invocation method to the second server communication object to execute against the specified remote object.

Applicants have amended claims 11, 30, 45, 53, 59, 65 to delete the term "respectively".

The claims require:

- (a) transmitting an invocation method specifying the remote method on one specified remote object from the client communication object to a first server communication object; and
- (b) transmitting, with the first server communication object, the received invocation method to the second server communication object to execute against the specified remote object, and nowhere does the cited Jones (col. 9: lines 29-32; col. 8: lines 10-15; col. 9: lines 52-59; col 9: lines 66 - col. 10, lines 2) teach or suggest these claim requirements.

The claims require a client communication object, a first server communication object and a second server communication object. The Examiner has indicated in Page 8 of the office action that col. 9, line 66 - col. 10, line 2 of the cited Jones discloses the claim requirement of transmitting, with the first server communication object, the received invocation method to the

second server communication object to execute against the specified remote object. Col. 9, line 66 - col. 10, line 2 of the cited Jones discusses a server invoking a method "using the received parameter argument values in RMI call" and the server returning "the result of the method invocation to client". Nowhere does the cited col. 9, line 66 - col. 10, line 2 of the cited Jones teach or suggest the claim requirements of transmitting, with the first server communication object, the received invocation method to the second server communication object to execute against the specified remote object.

Should the Examiner maintain the rejection of the claims, the Examiner is requested to indicate the following:

(A) Which element of the cited Jones is the first server communication object and which element of the cited Jones is the second server communication object?

(B) How the first and second server communication objects (if any such communication objects are indicated by the Examiner) teach or suggest the claim requirements of transmitting, with the first server communication object, the received invocation method to the second server communication object to execute against the specified remote object.

Claims 30, 45, 53, 59, 65 are patentable over the cited art for similar reasons provided above for the patentability of claim 11.

For the above reasons, claims 11, 30, 45, 53, 59, 65 are patentable over the cited art.

Claims 3, 22, 37

The Examiner has rejected claim 3 under 35 U.S.C. 103(a) as being unpatentable over Jones in view of Guthrie. The Examiner has rejected claims 22 and 37 for reasons similar to the rejection of claim 3.

Claim 3 depends on claim 1, wherein generating the at least one proxy object further comprises:

transmitting a request to instantiate a proxy object for one specified remote object and a specified proxy class;

generating the proxy object in response to the request;
calling an initialization method from the specified proxy class to add data from the specified remote object to the generated proxy object; and
returning the generated proxy object for use in accessing the remote object.

Nowhere does the cited Jones teach or suggest the claim requirement of calling an initialization method from the specified proxy class to add data from the specified remote object to the generated proxy object. The Examiner has indicated that col. 8, lines 50-55 and col. 9: lines 55-67 of the cited Jones discusses the claim requirements of calling an initialization method from the specified proxy class to add data from the specified remote object. Col. 8, lines 50-55 of the cited Jones discusses a stub compiler compiling the hash value into the local method. The Examiner appears to be indicating that the hash value of the cited Jones corresponds to the data from the specified remote object of the claim requirements. However, if that is so then should the Examiner maintain the rejection, the Examiner is requested to indicate:

(A) Which element of the cited Jones is the initialization method of the claim requirements;

(B) Where the cited Jones discusses that the initialization method adds the hash value from the specified remote object to the generated proxy object.

For the above reasons, claims 3, 22, 37 are patentable over the cited art.

Claims 10, 39, 44

Claim 10 depends on claim 7, wherein the client communication class includes code to handle error exceptions generated from the communication protocol class in response to executing remote methods on the plurality of proxy objects.

Col. 8, lines 55-58 of the cited Guthrie discusses that for each subject class methods, the software system determines the “exceptions” and “stores the information in JClass information”. However nowhere does the cited Guthrie teach or suggest that the claim requirement that the

code to handle error exceptions is generated in response to executing remote methods on the plurality of proxy objects.

Should the Examiner maintain the rejection of the claims, the Examiner is requested to indicate where the cited Guthrie teaches or suggests the claim requirement that the code to handle error exceptions is generated in response to executing remote methods on the plurality of proxy objects.

Additionally, there is no motivation in either the cited Guthrie or the cited Jones to combine the requirements of the cited Guthrie and the cited Jones to arrive at the claim limitations. The motivation provided by the Examiner that Guthrie's teachings would have provided the capability for enabling the client to uniquely identify the remote method without identifying the incorrect method is a general motivation that could lead to many different requirements and not the specific claim requirement that the client communication class includes code to handle error exceptions generated from the communication protocol class in response to executing remote methods on the plurality of proxy objects.

For the above reasons claims 10, 39, and 44 are patentable over the cited art.

Claims 13, 32, 47

Claim 13 depends on claim 5, wherein the client communication class includes code to handle exceptions related to communication between the client communication object and the server communication object.

Col. 8, lines 55-58 of the cited Guthrie discusses that for each subject class methods, the software system determines the "exceptions" and "stores the information in JClass information". There is no motivation in either the cited Guthrie or the cited Jones to combine the requirements of the cited Guthrie and the cited Jones to arrive at the claim limitations. The motivation provided by the Examiner that Guthrie's teachings would have provided the capability for enabling the client to uniquely identify the remote method without identifying the incorrect method is a general motivation that could lead to many different requirements and not the

specific claim requirement that the client communication class includes code to handle error exceptions generated from the communication protocol class in response to executing remote methods on the plurality of proxy objects.

For the above reasons claims 13, 32, and 47 are patentable over the cited art.

Claims 12, 31, 46

Claim 12 depends on claim 11, and further comprises:

executing, with the second server communication object, the remote method specified in the invocation method on the specified remote object;

returning, with the second server communication object, data generated in response to execution of the remote method on the specified remote object to the client communication object.

The cited Jones (step 506, 508) discusses that the server invokes the remote method using the received parameter arguments and returns the results. Nowhere does the cited Jones discuss a second server communication object as required by the claims. Should the Examiner maintain the rejection the Examiner is requested to indicate which elements of the cited Jones correspond to:

(a) the first communication object (found in base and/or intervening claims on which claims 12, 31, 46 depend); and

(b) the second communication object
of the claims requirements.

For the above reasons claims 12, 31, and 46 are patentable over the cited art.

Claim 16

Claim 16 depends on claim 15, wherein the server comprises a first server and wherein the called method comprises a remote method, further comprising:

transmitting, with the client, information indicating the called remote method and remote object corresponding to the proxy object subject on which the remote method is called;

receiving, with the first server, the information indicating the remote method and remote object to execute;

determining, at the first server, whether the indicated remote object is on a second server;
and

transmitting information on the indicated remote method and remote object to a second server for execution thereon if the indicated remote object for execution on the second server.

Nowhere does the cited Jones teach or suggest the claim requirement of client, a first server and a second server. The cited Jones discusses a client and a server. Should the Examiner the maintain the rejection of claim 16, the Examiner is requested to indicate which element of the cited Jones correspond to

- (A) the client
- (B) the first server
- (C) the second server.

For the above reasons claim 16 is patentable over the cited art.

Claim 17

Claim 17 depends on claim 16, and further comprises:

launching, with the client, an applet downloaded from the first server, wherein the applet calls the remote method on the proxy object corresponding to the remote object on the second server, and wherein the applet communicates the information indicating the remote method and remote object to the first server.

Nowhere does the cited Jones teach the claim requirements of

- (A) the client (from base/intervening claims on which claim 17 depends)
- (B) the first server
- (C) the second server.
- (D) the applet

as required by the claims. The cited Jones discusses a server and a client. Additionally, there is no teaching or suggestion of an applet in the Examiner cited col. 8, lines 13-15 of the cited Jones used to reject claim 17.

For the above reasons claim 17 is patentable over the cited art.

Claim 18

Claim 18 depends on claim 14, wherein the server comprises a first server and wherein generating the at least one proxy object further comprises:

receiving, with the first server, a method to instantiate a proxy object for a specified remote object;

determining, at the first server, whether the specified remote object is on a second server;

transmitting, with the first server, the method to instantiate the proxy object to the second server for execution thereon if the remote object is located on the second server.

Nowhere does the cited Jones discuss a client, a first server, and a second server as required by claim 18 and the base and/or intervening claims on which claim 14 depends. The cited Jones discusses a client and a server. In particular, nowhere does the cited Jones teach or suggest determining, at the first server, whether the specified remote object is on a second server and transmitting, with the first server, the method to instantiate the proxy object to the second server for execution thereon if the remote object is located on the second server.

For the above reasons claim 18 is patentable over the cited art.

Claim 19

Claim 19 depends on claim 18 and further comprises:

returning, with the second server, the instantiated proxy object to the first server to return to the client.

Col. 8, lines 10-15 of the cited Jones discusses that when client wishes to access a method implemented by a remote object on a server, the client uses a stub referencing remote object. The stub is downloaded from a server but can also be local to the client, or downloaded from another server. Nowhere does the cited Jones teach or suggest the claim requirement of returning, with the second server, the instantiated proxy object to the first server to return to the client.


For the above reasons, claim 19 is patentable over the cited art.

Conclusion

For all the above reasons, Applicant submits that the pending claims 1-65 are patentable over the art of record. Applicants have not added any claims. Nonetheless, should any additional fees be required, please charge Deposit Account No. 09-0466.

The attorney of record invites the Examiner to contact him at (310) 557-2292 if the Examiner believes such contact would advance the prosecution of the case.

Dated: June 13, 2005

By: 

Rabindranath Dutta
Registration No. 51,010

Please direct all correspondences to:

Rabindranath Dutta
Konrad Raynes & Victor, LLP
315 South Beverly Drive, Ste. 210
Beverly Hills, CA 90212
Tel: 310-557-2292
Fax: 310-556-7984